

**AMENDMENT AND PRESENTATION OF CLAIMS**

Please replace all prior claims in the present application with the following claims.

1. (Previously Presented) A method comprising:  
receiving at an electronic device an executable command specifying execution of an unidentified executable on first data without specifying which executable should be used for the first data;  
determining, at the electronic device, from metadata of the first data, a content type of the first data;  
determining to identify at the electronic device an executable using the content type; and  
determining to operate on the first data using the identified executable.
2. (Canceled)
3. (Previously Presented) A method as claimed in claim 1, wherein the command contains the metadata of the first data, and the metadata includes an identifier of the first data.
4. (Previously Presented) A method as claimed in claim 3, wherein the identifier identifies a node of a hierarchical nodular data structure stored at the electronic device.
5. (Currently Amended) A method as claimed in claim 4, wherein the command is a command selected from the group consisting of an Exec command, ~~and an~~ an Add command, a Copy command, a Delete command, a Get command, and a Replace command, and the identifier

is a uniform resource identifier contained within a source element corresponding to the node in the hierarchical nodular data structure.

6. (Previously Presented) A method as claimed in claim 1, wherein the command is received as extensible markup language code.

7. (Original) A method as claimed in claim 6, wherein the command is a SyncML command.

8. (Previously Presented) A method as claimed in claim 1, further comprising:  
determining to parse a content type of the first data based upon a node, wherein the content type of the first data is stored at the electronic device according to the node.

9. (Previously Presented) A method as claimed in claim 6, wherein the content type of the first data is stored at a node of a hierarchical nodular data structure.

10. (Previously Presented) A method as claimed in claim 9, wherein the node is a leaf node that identifies the content type of the first data.

11. (Canceled)

12. (Previously Presented) A method as claimed in claim 1, wherein the content type is determined by at least one of the value of a format element and the value of a type element associated with the first data.

13. (Previously Presented) A method as claimed in claim 1 further comprising determining to associate a plurality of different executables with each of a plurality of different content types.

14. (Previously Presented) A method as claimed in claim 1, wherein the executable is identified using the content type and a look-up table.

15. (Previously Presented) A method as claimed in claim 13, wherein the plurality of different executables are stored in the electronic device.

16. (Previously Presented) A method as claimed in claim 1, further comprising:  
before receiving the command specifying execution of the first data, determining to create or update a hierarchical nodular data structure at the electronic device.

17. (Withdrawn) A method, comprising:  
transferring code comprising a command to an electronic device, wherein the command specifies execution of an unidentified executable on first data stored at a first leaf node of a hierarchical nodular data structure;  
determining, from metadata of the first leaf node, a content type of the first data;  
identifying an executable using the content type determined from the metadata of the identified first leaf node; and  
operating on the first data, stored at the identified first leaf node, using the identified executable.

18. (Withdrawn) A method, comprising:

receiving re-usable code at an electronic device wherein the code comprises:

commands for creating at the electronic device a hierarchical nodular data structure, having leaf nodes and interior nodes, that comprises first data stored at a first leaf node; and a further command specifying execution of an unidentified executable on the first data stored at the first leaf node;

determining, from metadata stored at the first leaf node, a content type of the first data stored at the first leaf node;

identifying an executable using the content type determined from the metadata stored at the first leaf node; and

operating on the first data stored at the first leaf node using the identified executable.

19. (Withdrawn) An electronic device, comprising:

a memory configured to store first data and metadata of the first data;

a receiver configured to receive a command specifying execution of an unidentified executable on the first data; and

a processor configured to determine from the metadata of the first data, a content type of the first data, to identify an executable using the content type determined from the metadata, and to operate on the first data using the identified executable.

20. (Withdrawn) An electronic device as claimed in claim 19, wherein the receiver is further configured to receive set-up code, and the processor is configured to interpret the received set-up code to create a hierarchical nodular data structure, having leaf nodes and interior nodes, that comprises a first leaf node storing the first data.

21. (Withdrawn) An electronic device as claimed in claim 20, wherein the receiver is configured to receive the command in the set up code, and the processor is configured to interpret the command to determine, from the metadata of the first data, the content type of the first data.

22. (Withdrawn) A data structure embodied on a computer-readable medium, comprising: code identifying first data and specifying execution of an unidentified executable on the first data.

23. (Withdrawn) A data structure as claimed in claim 22, wherein the code further specifies the transfer of the first data to an electronic device.

24. (Withdrawn) A data structure embodied on a computer-readable medium, comprising: commands, execution of which create at an electronic device a hierarchical nodular data structure, having leaf nodes and interior nodes, that comprises first data stored at a first leaf node; and  
a further command identifying the first leaf node and specifying execution of an unidentified executable on the first data stored at the first leaf node.

25. (Withdrawn) A method, comprising: using a data structure as claimed in claim 22.

26. (Withdrawn) A method comprising: setting-up an electronic device using a data structure as claimed in claim 22.

27. (Withdrawn) A method comprising: re-using the data structure as claimed in claim 22, to set-up different electronic devices.

28. (Withdrawn) A server for storing and transmitting the data structure as claimed in claim 22.

29-33. (Canceled)

34. (Withdrawn) An electronic device, comprising:

means for storing first data;

means for receiving a command specifying execution of an unidentified executable on the first data;

means for determining, from metadata, a content type of the identified first data;

means for identifying an executable using the content type determined from the metadata; and

means for operating on the identified data using the identified executable.

35. (Withdrawn) A method, comprising:

providing code identifying first data and specifying execution of an unidentified executable on the first data and

transmitting the code.

36. (Withdrawn) A method, comprising:

transmitting commands for creating a hierarchical nodular data structure, having leaf nodes and interior nodes, that comprises first data stored at a first leaf node; and

transmitting a further command specifying execution of an unidentified executable on the first data stored at the first leaf.

37. (Withdrawn) A server, comprising:

a memory configured to store code identifying first data and specifying execution of an unidentified executable on the first data; and  
an interface configured to transmit the code.

38. (Withdrawn) A server as claimed in claim 37, wherein the operations further comprise setting up an electronic device.

39. (Withdrawn) A server as claimed in claim 37, wherein the operations further comprise re-using the code in setting up different electronic devices.

40. (Withdrawn) A server, comprising:

a memory configured to store commands, execution of which resulting in creation at an electronic device, of a hierarchical nodular data structure, having leaf nodes and interior nodes, that comprises first data stored at a first leaf node, and configured to store a further command identifying the first leaf node that specifies execution of an unidentified executable on the first data stored at the first node identifying the first leaf node that specifies execution of an unidentified executable on the first data stored at the first leaf node; and  
a transmitter configured to transmit the stored instructions.

41. (Previously Presented) A computer-readable storage medium carrying one or more sequences of one or more instructions which, when executed by one or more processors, cause an apparatus to at least perform the following steps:

receiving an executable command specifying execution of an unidentified executable on first data without specifying which executable should be used for the first data;  
determining, from metadata of the first data, a content type of the first data;  
determining to identify an executable using the content type; and  
determining to operate on the first data using the identified executable.

42. (Canceled)

43. (Withdrawn) A method, comprising:

receiving a first command at an electronic device, the first command specifying creation of a leaf node in a hierarchical data structure, and identifying first data to be stored at the leaf node and metadata indicating a content type of the first data;  
creating the leaf node at the electronic device;  
receiving a second command, at the electronic device, that specifies execution of an unidentified executable on the first data stored at the created leaf node;  
determining, from the metadata, the content type of the first data;  
identifying an executable using the content type determined from the metadata; and  
operating on the first data using the identified executable.

44. (Withdrawn) An electronic device, comprising:



a receiver configured to receive a first command at an electronic device, the first command specifying creation of a leaf node in a hierarchical data structure, and identifying first data to be stored at the leaf node and metadata indicating a content type of the first data; and a processor configured to create the leaf node at the electronic device, wherein the receiver is further configured to receive a second command that specifies execution of an unidentified executable on the first data stored at the created leaf node, and the processor is further configured to determine, from the metadata, the content type of the first data, to identify an executable using the content type determined from the metadata, and to operate on the first data using the identified executable.

45. (Previously Presented) A method as claimed in claim 1, wherein the command excludes information of the content type of the first data.

46. (Previously Presented) A method as claimed in claim 16, further comprising: before receiving the command specifying execution of the first data, determining to create the node or a sub-tree including the node in the hierarchical nodular data structure at the electronic device.

47. (Previously Presented) A method as claimed in claim 12, wherein the executable is identified using a look-up table and the at least one of the value of the format element and the value of the type element.

48. (Previously Presented) A method as claimed in claim 1, wherein the first data includes media content data.

49. (Previously Presented) An apparatus comprising:

at least one processor; and

at least one memory including computer program code for one or more programs,

the at least one memory and the computer program code configured to, with the at least one processor, cause the apparatus to perform at least the following,

receive an executable command specifying execution of an unidentified executable on

first data without specifying which executable should be used for the first data,

determine, from metadata of the first data, a content type of the first data,

determine to identify an executable using the content type, and

determine to operate on the first data using the identified executable.